A Step Forward in Semi-automatic Metamodel Matching: Algorithms and Tool

José de Sousa Jr[1], Denivaldo Lopes[1], Daniela Barreiro Claro[2], and Zair Abdelouahab[1]

[1] LESERC, Electrical Engineering Department
Federal University of Maranhão, Av. dos Portugueses, s/n São Luís - MA - Brazil
{jgeraldo,dlopes,zair}@dee.ufma.br
http://www.leserc.dee.ufma.br/
[2] Distributed Systems Laboratory (LaSiD), Computer Science Department
Federal University of Bahia, Av. Adhemar de Barros, s/n Salvador - Bahia - Brazil
dclaro@ufba.br
http://www.lasid.ufba.br/

Abstract. In recent years the complexity of producing softwares systems has increased due the continuous evolution of the requirements, the creation of new technologies and integration with legacy systems. When complexity increases the phases of software development, maintenance and evolution become more difficult to deal with, i.e. they became more subject to error-prone factors. Recently, Model Driven Architecture (MDA) has made the management of this complexity possible thanks to models and the transformation of Platform-Independent Model (PIM) in Platform-Specific Models (PSM). However, the manual creation of transformation definitions is a programming activity which is error-prone because it is a manual task. In the MDA context, the solution is to provide semi-automatic creation of a mapping specification that can be used to generate transformation definitions in a specific transformation language. In this paper, we present an algorithm to match metamodels and enhancements in the MT4MDE and SAMT4MDE tool in order to implement this matching algorithm.

Keywords: Metamodel matching, Algorithm, Mapping specification.

1 Introduction

The software production process in MDA is based on models and model transformation. In the MDA context, some research and proposals of transformation languages are available in the literature [1,2,3,4] or in the form of products.

However, the manual creation of transformation definition remains a programming activity that is tedious and error-prone. Thus, an approach that makes the automatic generation of transformation definition possible can leverage the MDA domain.

The task of creating transformation definitions between models is preceded by a mapping specification that consists of searching elements which are semantic and/or syntactic equivalent (or similar) between the target and source metamodel. However, the creation of mapping specification between two metamodels is not an easy task, because metamodels are generally created with different goals and by different development

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 137–148, 2009.

teams. This leads to a structural and semantic distance among metamodels that are called *gaps* among metamodels [5].

The manual creation of mapping specification is an activity that becomes more difficult and time consuming as the relationships between metamodels are more complex, i.e., more effort is required to find them.

The creation of mapping specification is also an important task in the fields of database such as database integration, E-business and data warehousing. The correspondence between *schemas* in different databases is called *schema matching* and it plays a similar role to the mapping specification in MDA. The proposed solution to simplify the determination of these correspondences in MDA is based on the use of a semi-automatic matching tool, i.e., automatically detecting what elements are corresponding and interacting with a user so as to validate the suggestions made by the *matching algorithm* [6].

In this paper the problem of determining mapping specification is investigated. An algorithm for detecting correspondences presented in [7] and applied to *schema matching* in the field of database systems is enhanced and adapted to be used in field of MDA. An implementation of this algorithm was done using a tool called Semi-Automatic Tool for Model-Driven Engineering (SAT4MDE) [6].

This paper is organized as follows. Section 2 presents an overview of the technologies and concepts involved in the software development process using a model driven approach. Section 3 presents an approach and an algorithm to make the semi-automatic mapping specification possible. Section 4 describes the tool SAM4MDE and its extension to insert the algorithm to generate mapping specification. Section 5 presents tests and the evaluation of the results obtained by the semi-automatic generation of mapping specification. Section 6 contains conclusions and the future directions of this research work.

2 Background

The aim of this section is to present some technologies and tools used in the context of Model-Driven Engineering (MDE) and consequently related them to our research work.

2.1 Model Driven Engineering

Model Driven Engineering (MDE) is an approach that has models as main focus in order to provide benefits such as cost reduction and increased quality of software products. The relevance of models in MDE does not only consist of documenting software systems, but these are formal models that can be understood by computers, i.e. they contain information that can be easily manipulated by a computer. The model manipulation is done through model transformation, a technique that consists in obtaining models from another model. Any modification made in the software is done inside models and the transformation is repeated so as to diffuse the changes.

Model-Driven Engineering is a recent approach and requires sophisticated formalism, i.e. stable technique and tools that allow the creation of consistent software, following a methodology for applying MDE. Some initiatives have been proposed in the